future internet

MDPI

Article

Deep Learning-Based Symptomizing Cyber Threats Using Adaptive 5G Shared Slice Security Approaches

Abdul Majeed [1], Abdullah M. Alnajim [2,*], Athar Waseem [1], Aleem Khaliq [1], Aqdas Naveed [1], Shabana Habib [2], Muhammad Islam [3] and Sheroz Khan [3]

[1] Department of Electrical Engineering, Faculty of Engineering and Technology, International Islamic University, Islamabad 44000, Pakistan; abdul.majeed@iiu.edu.pk (A.M.); athar.waseem@iiu.edu.pk (A.W.); aleem.khaliq@iiu.edu.pk (A.K.); anaveed@iiu.edu.pk (A.N.)
[2] Department of Information Technology, College of Computer, Qassim University, Buraydah 51452, Saudi Arabia
[3] Department of Electrical Engineering, Onaizah College of Engineering and Information Technology, Onaizah Colleges, Qassim 56447, Saudi Arabia; m.islam@oc.edu.sa (M.I.); cnar32.sheroz@gmail.com (S.K.)
* Correspondence: najim@qu.edu.sa

Abstract: In fifth Generation (5G) networks, protection from internal attacks, external breaches, violation of confidentiality, and misuse of network vulnerabilities is a challenging task. Various approaches, especially deep-learning (DL) prototypes, have been adopted in order to counter such challenges. For 5G network defense, DL module are recommended here in order to symptomize suspicious NetFlow data. This module behaves as a virtual network function (VNF) and is placed along a 5G network. The DL module as a cyber threat-symptomizing (CTS) unit acts as a virtual security scanner along the 5G network data analytic function (NWDAF) to monitor the network data. When the data were found to be suspicious, causing network bottlenecks and let-downs of end-user services, they were labeled as "Anomalous". For the best proactive and adaptive cyber defense system (PACDS), a logically organized modular approach has been followed to design the DL security module. In the application context, improvements have been made to input features dimension and computational complexity reduction with better response times and accuracy in outlier detection. Moreover, key performance indicators (KPIs) have been proposed for security module placement to secure interslice and intraslice communication channels from any internal or external attacks, also suggesting an adaptive defense mechanism and indicating its placement on a 5G network. Among the chosen DL models, the CNN model behaves as a stable model during behavior analysis in the results. The model classifies botnet-labeled data with 99.74% accuracy and higher precision.

Keywords: PACDS; VNF; vCTS; interslice; intraslice; NWDAF; 5G network; slice security

Citation: Majeed, A.; Alnajim, A.M.; Waseem, A.; Khaliq, A.; Naveed, A.; Habib, S.; Islam, M.; Khan, S. Deep Learning-Based Symptomizing Cyber Threats Using Adaptive 5G Shared Slice Security Approaches. *Future Internet* **2023**, *15*, 193. https://doi.org/10.3390/fi15060193

Academic Editors: Mario Di Mauro, Francesco Pascale, Marco Tambasco and Ivan Serina

Received: 5 March 2023
Revised: 20 May 2023
Accepted: 24 May 2023
Published: 26 May 2023

1. Introduction

The phenomenon of KPIs in 5G networks specifies the functionality from the access level to the core network, to supporting critical network functions (NFs) [1,2]. Network slicing has been developed as a special logical and virtualized network on top of the multi-domain physical structure, enabling the coexistence of many verticals "Slices" over the same infrastructure. Each network slice has functional resources that are either dedicated, shared, or both; separated from other network slices; and have useful potential based on the requirement of each slice. Network slicing can be incorporated into the core network (CN) using both software defined networking (SDN) and network function virtualization (NFV). NFV and SDN demonstrate how software functions can be connected to share physical resources and can be operated in a virtual environment.

Among new network security challenges, 5G slice security has not yet been explored [3]. A rule-based slice-service-type (SST) security system is generally initiated on 5G

networks. The physical resources of the network are shared among users or enterprises and can be managed in a secure environment using virtualization. The process of the slice initiating or embedding with resources during its execution cycle causes many loopholes in configuration, malware injection, or telnet session misconfiguration, which can help hackers to compromise the shared resources in a 5G environment. Moreover, legitimate tenants (users and enterprises) could facilitate the penetration of attacks while using suspicious third party services in a 5G network [4–8]. The auto-negotiation and intra-communication of shared resources with distinct security measures is also an emerging phenomenon in 5G slicing [9,10]. In the context of 5G slice resiliency and security control, to what extent can shared network resources can be managed, preserved in quarantine, and kept safe from paired suspicious resources? Hence, the attack vectors are multi-directional and can be dispersed in networks more swiftly [8,11]. To mitigate or suppress such attack vectors before activation, much effort has taken place in the field of AI using DL security modules, following proactive security schemes to optimize slices [12,13]. Further, the placement of security modules or NFs is a challenging task. Points of presence (PoPs) of security scanners and their integration can reduce the operating expenditure (Opex) and provide better QoS in a 5G virtual infrastructure. Virtual security as a service (Vsaas) is an upcoming challenge, and its integration and placement in virtual slices can facilitate VNF operators and provide a reliable solution to the dynamic security deployment of 5G and B5G networks [14,15].

When talking about speedy networks, e.g., 5G networks (high data rates and quick convergence) can encounter AI multi-vector attacks. While traditional defense systems fail to block such attacks because of their obsolete and stationary architecture [16], conventional security systems become overloaded and drop packets due to large data. Additionally, parallel techniques, e.g., FPGA and ASIC, have been adopted to deal with AI attacks, but these cutting-edge technologies are bound by hardware limitations. For zero-day attack detection, DL models have been proven as the most superior forms and make a significant contribution to 5G and Beyond 5G (B5G) infrastructure security and robustness. Designing a diversified security system with an advanced architectural approach is more challenging than ever before. Considering essential security measures to safeguard the cyber world through the strong, effective, and adaptive security steps can prove more sustainable and proactive in the real-time identification of attack vectors in modern networks with logical reasoning.

For zero-day attack detection, we suggest an AI-based adaptive security model using adaptive defense systems (ADSs), as shown in Figure 1, focusing on a modular approach to design advanced PACDSs, which can help to develop and deploy the DL security modules and their placements on PoPs, which are defined by the proposed KPIs for 5G network security. This supervisory approach can help to monitor the NetFlow on shared slices more efficiently and rapidly and to empower network security systems to be smarter and more tactically secure in regard to intelligent attacks.

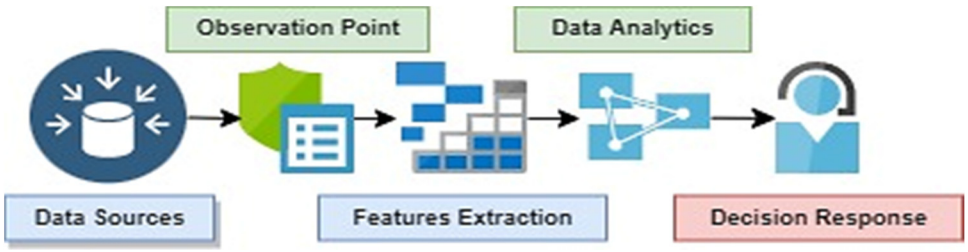

Figure 1. Adaptive defense systems—ADS [17].

This paper mainly contributes to the design of a CTS module, such as VNF. It can be deployed on interslice and intraslice communication channels to deal with security challenges in a shared environment, and furthermore, can identify the places for the security scanners' PoPs among slices and associated VNFs. We seek to deploy the CTS module on multiple streams of the 5G network for threat detection and channel optimization.

Using isolation as a tool, we utilize the proposed "KPIs" to secure 5G networks from compromised 5G tenants. This effort will ensure the stable E2E QoS among slices and their VNFs in a shared architectural system. The research objective is to create an efficient defense mechanism that not only defends network services but also deals with the issue of security service placement on 5G networks. The suggested architecture maintains a degree of stability in terms of QoS, budgetary costs, and security as a service. KPIs-based network analysis also assists an operator in mounting a NetFlow-based virtual security scanner on desired logical channels or in a space between multiple network domains. For tactical and partial security deployment, it is necessary to define the logical partition of the network, whilst considering communication channels and NFs. For logical partitions, the following suggested KPIs are more helpful for mounting data checkpoints on:

- Inter-slice communication channels;
- Intra-slice communication channels;
- 5G and tenants' network (2G, 3G, 4G) communication channels.

The security PoPs are logical blueprints for mounting checkpoints for backhaul and fronthaul security to the 5G core. The 5G network is a service-based architecture (SBA) system that uses service-level agreements (SLAs) with various tenants. The 5G supports the sharing of services used by operators. The shared service can be compromised using a third-party network service in a shared slice. The proposed security schemes, e.g., the design of the CTS module, the logical partition of the network, and defined security PoPs, provide real-time threat detection in 5G network streams. Network data analytic function (NWDAF) [18–20], a novel feature, has been used to assist the security program integration of a CTS module on 5G networks for adaptive defense. Moreover, the train and test models used to improve classification accuracy contrast with past efforts based on experimental work. The virtualized CTS (vCTS) as a security module is used to generalize a higher accuracy rate and a quicker response time in order to narrow down the research gap in past contributions in anomaly detection and make an additional effort in the placement of virtualized security modules.

2. Materials and Methods

The DL-based security module design, configuration, and hyper-parameter selection has been considered to provide the maximum classification and detection accuracy in anomaly detection for better performance and designated throughput. Considering the two-tier anomaly detection method, we followed a modular approach to design a virtualized security module. The security module is used to symptomize suspicious data as an anomaly. The proposed CTS module behaves as an intelligent DL agent that can provide sufficient NetFlow information to analyze any attack's inception. The recommended KPIs serve as the blueprint for the tactical deployment of security modules in 5G architecture. It is an additional approach to 5G architecture by isolating the network logically in order to deploy the proactive and adaptive defensive mechanism for committed E2E QoS. The suggested process steps of the methodology for PACDS are described and shown in Figure 2, which are taken into account in order to design and deploy the DL security module in the defined PoPs of the network for the best PACDS. Additionally, the CTS module is able to provide sufficient threat information to optimize slices and their VNFs in a shared environment. The methodological domain follows the suggested process steps, which are:

1. Information—vertical industry.
2. NetFlow accounting—security scanners.
3. Data preparation for input feature vectors.
4. DL security module design.
5. Model analysis based on symptoms classification.
6. KPIs-based security module placement as VNF in 5G communication channels.

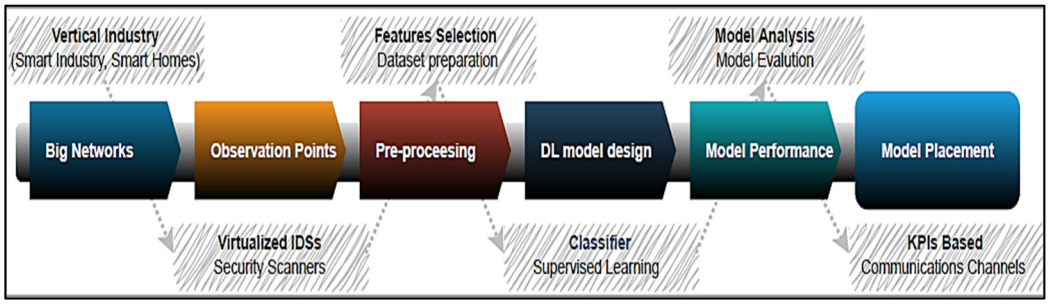

Figure 2. Proactive and adaptive cyber defense system design flow-graph.

2.1. Input Feature—Vector Selection

The process of meaningful feature selection and utilizing these features to train an ML model is mandatory when dealing with large amounts of data. ML supports numerous arrangements for data handling, comprising NetFlows and pcap packet-capture data [21]. Scenario 11 of the CTU-13 dataset has been selected for threat symptomizing. The dataset holds three labels, namely "background", "Normal", and "Botnet". The dataset contains labeled data with multiple real botnet attack vectors rather than simulated ones, which have been used for the training and testing of the opted models. Before model evaluation, it is necessary to pre-process and normalize the selected data as filtered input vectors for the DL model to achieve better performance. The data columns have many values, such as source IP addresses and multi-range data columns, which cannot be transformed into suitable input data vectors, and also low-scope or parsing value data columns, which cause delays in model execution. All such data columns are removed from the dataset to decrease the data size. The remaining columns, such as "Dir" and "Proto" with low cardinality but retaining categorical values, are considered for pre-processing. The "Dir" column has 6 categories, and the "Proto" column has 13 categories. The categorical columns are "one-hot" encoded to binary input vectors. Encoding is used to transform the categorical column into single-value binary columns. In addition, the dataset with large-scale data columns, such as "Dur", "Tot-Pkts", "Src-Bytes", and "Tot-Bytes", may exhaust memory or cause an interruption in the training process. This problem has been addressed via the normalization of such columns in a range close to the "0" and "1" values, and the large-scale data columns are listed with min–max values, as seen in Table 1.

Table 1. Large-scale data columns.

	Min Value	Max Value
Dur (secs)	0.0	971.284058
Tot-Pkts (count)	1	498,932
Tot-Bytes (count)	60	480,143,707
Src-Bytes (count)	0	22,872,870

Moreover, the "sTos" and "dTos" columns carry null values and have been removed from the data. Indeed, ML models tend to acquire majority-class features and have a penchant for over-fitting. During training cycles, models have achieved accuracy mostly in the majority class. To avoid this data imbalance, oversampling is used. Here, the Synthetic minority oversampling technique (SMOTE) is used to add multiple minority-class sample groups, as represented in Figure 3. The default minority class samples induction ratio is k = 5, which designates a close neighbor of the same class. Near the boundary between minority samples and their closest neighbors, synthetic samples are produced.

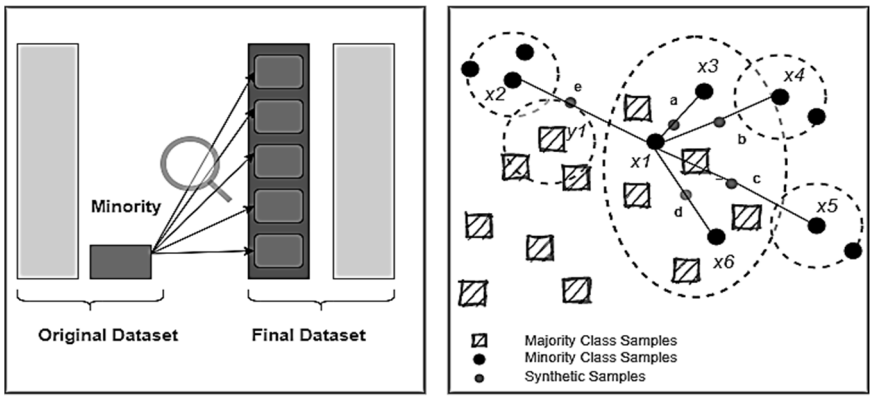

Figure 3. Synthetic minority oversampling technique—SMOTE [22].

The target column "label" contains the output labels of normal and anomalous classes. We mapped the label "0" for "Background and Normal" data and "1" for "Botnet" data. We defined Class "0" as a normal class and Class "1" as an anomalous class.

2.2. Analytical Engine Selection

Recently, researchers have been emphasizing DL techniques to build security models, such as DL models of a generative or discriminative nature. Recurrent neural networks (RNN) and convolutional neural networks (CNN) are discriminative models, which are mostly adopted for DL security model design. RNNs models classify data with discriminative power when input data correlates with the models' output labeled sequence explicitly. The RNN model is one of three types: simple RNN, long short-term memory unit (LSTM), and gated recurrent unit (GRU) [23]. The LSTM is one of the most popular RNN systems as well as the most effective in response to temporal data, e.g., speech recognition. The model uses sequencing with two LSTMs, well known as "Bi-LSTM", being fed in both forward and backward directions for the process cycle. The CNN model contains one or more "convolutional" and "pooling" layers in the form of an array to construct a multilayer neural network [23–25]. For spatial and temporal pattern classification, a hybrid approach combines LSTM and CNN functionality. Conv-LSTM first picks up spatial features from visuals and images, while simultaneously honing temporal features [26]. For the DL model design, we have to evaluate Bi-LSTM, CNN-LSTM, and CNN models for NetFlow analysis on a chosen dataset. The performance of each DL model has been examined using selected configurations and hyper-parameters, as given in Table 2. Additionally, the results are investigated thoroughly for the optimum model selection among the three. The outcomes of each model have been evaluated based on metrics, e.g., recall, precision and F1, and the confusion matrix (CM), that help to assess the best-fit DL model, which can classify the temporal and spatial data features on 5G networks [27,28].

Table 2. Hyper-parameters for CTS module evaluation.

Hyper-Parameters	Bi-LSTM	CNN-LSTM	CNN
Input Feature Size	24	24	24
Batch Size	10,000	10,000	10,000
Layers Size	10	16-14-12	16-14
Activation Function	Leaky-ReLU	Leaky-ReLU	Leaky-ReLU
Drop-Out	0.1	0.1	0.1
Output Layer Size	2	2	2
Output Activation Function	Softmax	Softmax	Softmax

2.3. Analytical Procedure

In this section, the analytical procedure has been described as shown in Figure 4. The process of analysis starts with the partition of the dataset into two sets. The "Training set"

is used to train the model, and later, the "Test set" is used to evaluate the model's accuracy. The CTS module as the analysis engine (AE) comprises a DL framework library that is used to design and develop a DL model. The model evaluation determines the overall accuracy of the model on unobserved data. In NetFlow analysis, the CTS module observes suspicious data or any attack data that pass through security checkpoints. The CTS module can be virtually initiated by the operator if any services or links seem to be compromised. As an action, the CTS module stamps the suspicious data and forwards the symptomized labeled data to the next level module for deep packet inspection (DPI). DPI determines the contents of packets with symptoms and categorizes the attack vectors for future analysis and network safety. NetFlow data without suspicious symptoms are considered normal data. The primary focus is on the first level VNF as a CTS security module and its multi-tier deployment on a 5G network.

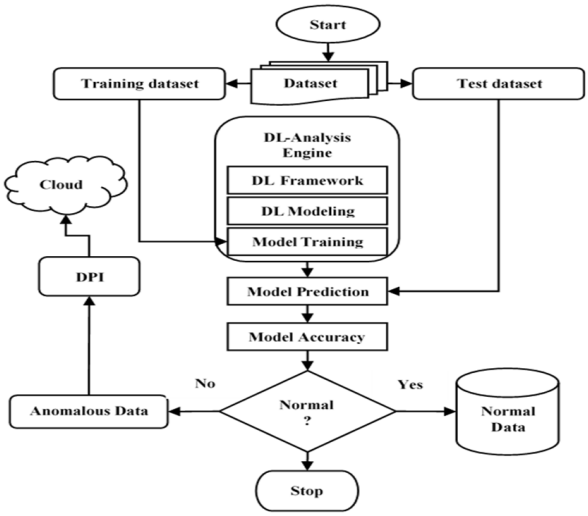

Figure 4. CTS module analysis flow-chart.

2.4. Hyper-Parameter Selection

The following configuration and hyper-parameters are defined to design and evaluate the CTS module, shown in Table 2.

The given hyper-parameters and configuration have been chosen to decide the performance of the best-fit model. The model input feature size comprises 24 feature vectors (Fv). The batch size is fixed at 10,000 Fv per second. Previously, the ReLU activation function was used for cross-validation among neural network layers. The ReLU is notorious for the problem of dying nodes in neural networks. ReLU was replaced with the optimized leaky ReLU activation function for cross-validation among network layers. Drop-out regularization has been tuned to 0.1 for nodes released from a neural network. For binary classification, the output layer size is set to 2. The Softmax function is typically used in neural network models as an output layer activation function; it can convert logits into probabilities. On the configuration and hyper-parameters selected, the models are trained and tested on certain epochs for classification accuracy in "Normal" and "Botnet" classes, respectively. For neural network design, the Bi-LSTM size is fixed to 10 filter layers for both forward and backward sequencing, the CNN-LSTM model has convolution layer sizes of 16–14, and the temporal input layer size is set to 12. For CNN, the same 16–14 layer size is adopted. Moreover, Python 3.10 is used as a DL framework to construct and evaluate all models. Simulation and experimental work was carried out on a Google-Collaboratory Notebook.

2.5. Multi-Tier Approach for Placement of vCTS Module

Transport layer security (TLS) and IP firewalls do not offer complete protection from intrusions. Assuring a security level where data cannot be breached requires fine-grained multi-tier security, where SLA can be employed safely at the access to core levels. A KPI-

based defense mechanism is a novel approach for deploying security modules, as depicted in Figure 5. The proposed defense system design is based on the 5G–SBA data analytic service "NWDAF". In a shared slice environment, virtualized (vNWDAF) can be used as a data analyzer with associated VNFs, which provides the ability to collaboratively work with third-party NFs. This ability of vNWDAF provides an advantage when coupling vCTS security modules to a slice in a shared environment. This approach allows an operator to query an event analysis using vCTS to examine intraslice data activity through VNFs communication on a shared slice. vCTS deployment at this level not only provides NetFlow monitoring but also raises an alarm or identifies symptomized suspicious activities in real-time. Additionally, it enables the safety of core slices and their associated NFs, particularly when these are shared with the tenants during SLA compliance. Shared services can be endangered via third-party network services in a virtual environment. The tenant can use both 5G and third-party wireless services. The compromised third-party services used by the legitimate 5G user can expose 5G slice services or cause bottlenecks or halt the network. Compromised 5G slice services can be dangerous to other 5G slices. However, we have outlined this novel challenge using predefined KPIs for the vCTS module deployment at logically demarcated PoPs on the 5G architecture. For cyber threat symptomization, the CTS module is placed as virtualized security scanner for data analysis on the 5G network:

- Intra-slice (NFs): the CTS module behaves as a core security scanner as NF and can demand the event analysis from 5G NFs.
- Core filter (backhaul): the CTS filter can monitor the backhaul channel between the core and edge depicting the main to regional topography of the 5G system.
- Slice filter (front-haul): at this level, the CTS acts as a slice security scanner to safeguard individual slices with a distinct slice-ID.
- Inter-slice filter (multiple slices): inter-slice filtering is applied across 5G slices, where each slice is monitored by individual vCTS to safeguard from compromised tenants (using third party network services).

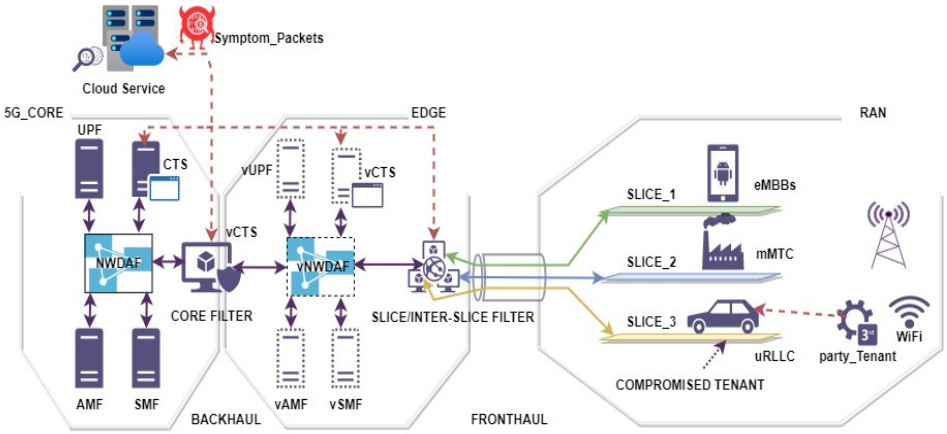

Figure 5. Multi-tier placement of vCTS modules on security PoPs.

The novelty of our approach is to enhance security at 5G verticals in contrast to past efforts based on physical rather than virtual infrastructure, which only deals with security at the RAN level. The deployment of the CTS module not only at the 5G RAN level but to promote the strategy from access to core level using the advantages of the SDN-NFV plane to initiate a security module on demand with multi-tier scanning capabilities in 5G communication channels in virtual slices (shared twins) in virtualized environments based on proposed KPIs.

3. Related Work

Software-defined networking refers to the physical separation of the network control plane from the data plane. The control plane controls various network devices, and the data

plane is used to route network data. The specialty of SDN is the steering of the data plane without engaging the underlying physical architecture, whereas NFV is used to integrate various VNFs, such as load balancers, databases, and security modules. NFV refers to a high-level reference plane, as shown in Figure 6. The management and orchestration (MANO) plane has been used for virtual infrastructure management and orchestration [29–31].

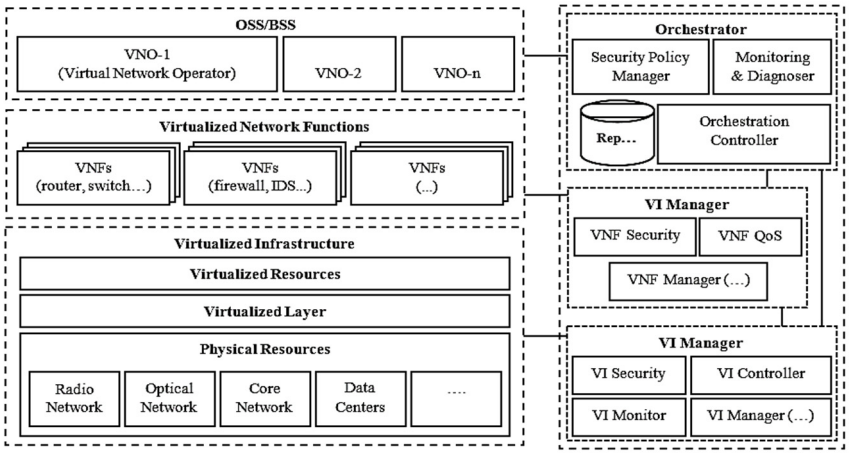

Figure 6. High-level MANO plane using NFV architecture [29,30].

Based on the high-level MANO plane, the two-level VNF system was recommended by researchers [29]. The first-level VNF ASD (anomaly symptom detection) analyzes the NetFlow streams received from the flow collector, which is part of the 5G radio access network (RAN). ASD symptomized or labeled the suspicious data features using supervised and semi-supervised learning, based on sliding window operation. At a later level, symptomized packets were forwarded to the second VNF NAD (network anomaly detection) for deep packet inspection (DPI) to the evolved packet core (EPC), as shown in Figure 7. The performance of numerous CPU- and GPU-based DL frameworks was examined to deal with big data [30]. The authors of [29] further extend their work to a two-level botnet symptomizing approach using a CTU-13 dataset.

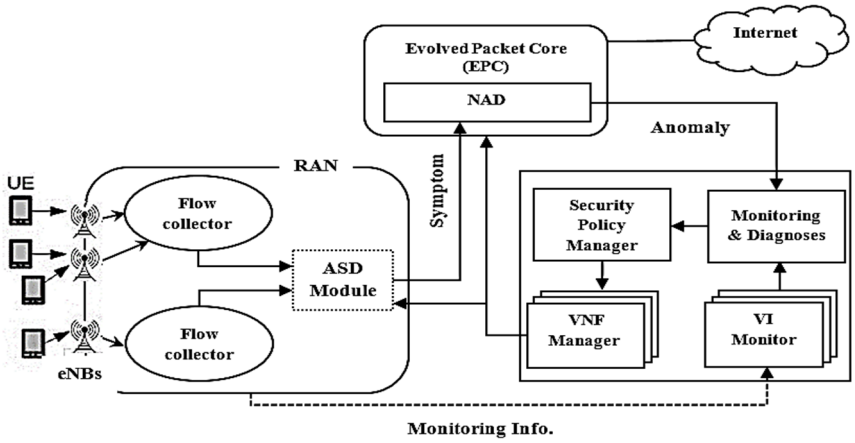

Figure 7. Symptoms detection system—two-tier approach [29].

On botnet labeled data, a deep belief network (DBN) and stack auto-encoder (SAE) were scored, namely F1-score—0.8940; recall—0.9934; and precision—0.8126 [29]. In addition, the same group of researchers fine-tuned their model and examined the NetFlow before sending it to network slices, the suspicious data were stripped by substituting suspicious data. For an integrated clinical environment (ICE), an SDN-NFV architecture was used with a few improvements in the model with a metric recall score of 0.9954 and a metric precision score of 0.9537 [31]. The sliding-window approach faces multi-class (fixed-size)

window problems. Sliding-window techniques often justify pairs of flows as an anomaly, even though they display no attack symptoms. In the time series data, this gap has been considered through two-dimensional image input data "image with time frame t", shifting the human activity classification (HAC) problem to image segmentation. Inspired by the two-level VNF approach [29], a DL residual U-Net (CNN) module was designed by another group of researchers. The residual U-Net with a focal Tversky loss function performed with the metrics: precision = 0.138; recall = 0.7257; and F2 = 0.4. The authors claimed that the suggested technique was an alternative solution to the sliding window multiclass classification problem in the self-adaptive models for anomaly detection [32]. NWDAF is a 5G service (SBA) that provides the ability to mount an ML model of one's own choice for network analysis. It has been deployed for network load optimization and suspicious data analysis, in which it has performed dual tasks, first as an "information behavior analyzer" and, additionally, to "percept the fictitious data classification" on suspicious datasets [20].

4. Results

In contrast to the previous two-level anomaly detection approach at the 5G RAN (Figure 7), we extended our method to a multi-tier security approach across the 5G network (Figure 5). The vCTS module can symptomize the suspicious NetFlow data from communication channels at the Core, Edge, and RAN levels. The vCTS can be virtually initiated to optimize and secure the network slices from external attacks. We evaluated three DL models to deal with both temporal and spatial data models in discriminative manners on an opted dataset. Moreover, the chosen dataset was used to evaluate Bi-LSTM, CNN-LSTM, and CNN models rather than a single model in order to consider all data aspects of input for the future meta-data nature 5G slices. For the optimum model selection in anomaly detection, the performance of all opted models has gone through exhaustive evaluation and discussion. The models have been trained appropriately after some feature engineering and pre-processing for suitable input feature vectors. The dataset has split into training sets (80%), presenting "X_train" and "X_test", in order to validate models, and test sets (20%), which are used to evaluate two classes, namely "Normal" and "Botnet", using bi-label classification for cyber threat symptomizing. Models were trained to a maximum of 20 iterations for stable output. "Train-Test loss" and "Train-Test accuracy" of each model were analyzed. The behavior of each DL model was observed and compared to all model outcomes. The analysis of each model is described, considering the metrics recall, precision, F1, and confusion matrix.

4.1. Bi-LSTM Model Behavior Analysis

Figure 8 shows the behavior of the Bi-LSTM in the "loss vs. accuracy" during the model train–test. It determines the model attitude by showing a fast reduction in loss (improved accuracy) until the tenth epoch and observing a gradual reduction until the last epoch.

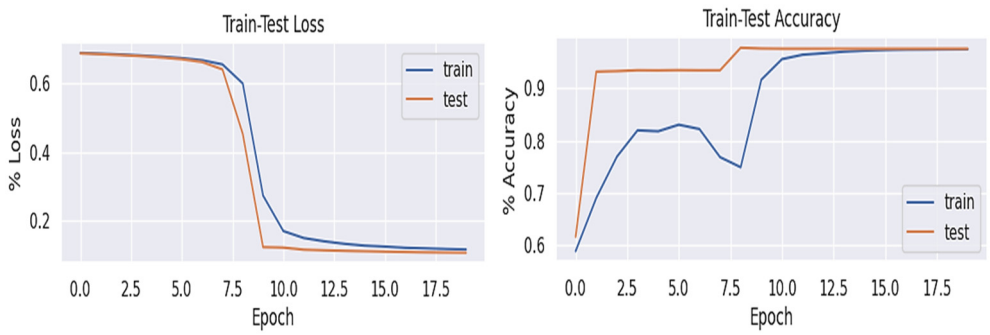

Figure 8. Bi-LSTM train–test loss vs. accuracy.

The comparison of "Normal' and 'Botnet" class samples are represented in the confusion matrix. Bi-LSTM classification results are displayed in Figure 9. From 19,709 valid labels of "Normal class", out of which 18,818 are correctly categorized as normal samples with a 95.6 percent accuracy. In the "Botnet class", the model predicted 19,874 sample as botnet out of 19,926 true labels. Malicious labels were classified with 99.7% accuracy. The total number of predictions from each class categorized successfully is shown in the confusion matrix.

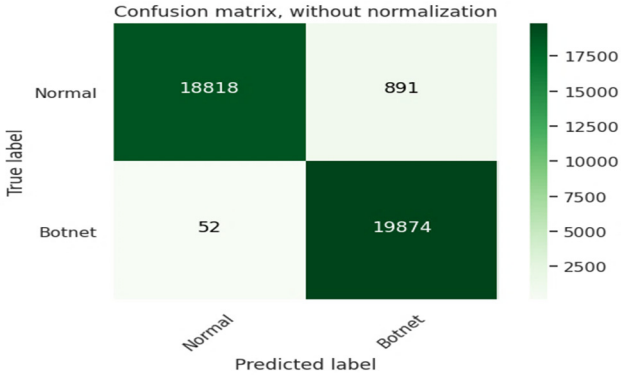

Figure 9. Confusion matrix—Bi-LSTM.

The recall of the normal class is 95.6%, and 4.7% of unseen suspicious data was found by the Bi-LSTM during the model test. In contrast, the botnet's detection accuracy is 99.7%. According to Table 3, the model performs quite well in terms of accuracy with an F1-score of 98%.

Table 3. Classification score—Bi-LSTM.

	Precision	Recall	F1-Score	Support
Class 0	1.00	0.95	0.98	19,709
Class 1	0.96	1.00	0.98	19,926
Accuracy			0.98	39,635
Macro Avg.	0.98	0.98	0.98	39,635
Weighted Avg.	0.98	0.98	0.98	39,635

4.2. CNN-LSTM Model Behavior Analysis

The CNN-LSTM model train–test "Loss vs. Accuracy" is described in Figure 10. Until the eighth epoch, the performance of the CNN-LSTM model shows a gradual drop in the loss. Initially, some variations have been recorded in model accuracy up to the sixth iteration, later on, the model converges smoothly and is found to be stable in the final epoch.

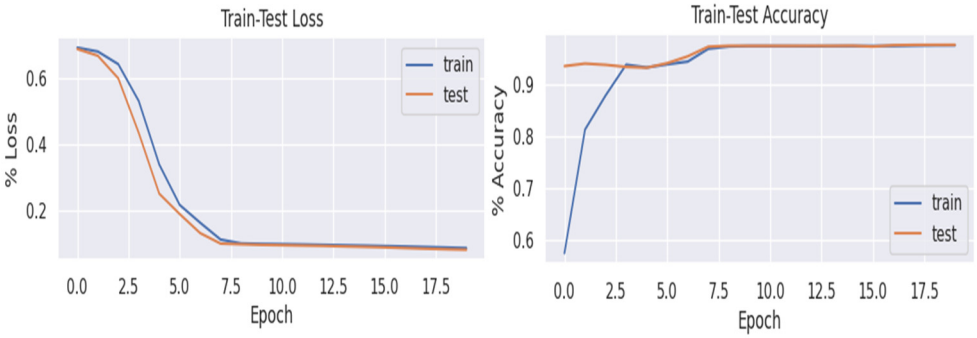

Figure 10. CNN-LSTM train–test loss vs. accuracy.

The classification of the labels of the "Normal" and "Botnet" classes produced by the CNN-LSTM model is depicted in the confusion matrix, as shown in Figure 11. The 18,890

out of 19,709 true labels were classified in the "Normal" class at 95.84%. In contrast, the model predicts malicious data with 99.74% accuracy as anomalies out of 19,926 true labels.

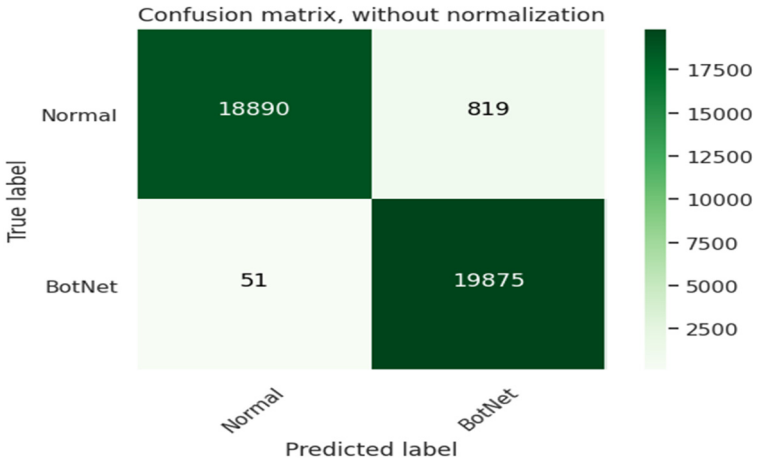

Figure 11. Confusion matrix CNN-LSTM.

The CNN-LSTM model performance determines the recall score of 95%, where 4.42% of unobserved labeled data are identified as suspicious. The accuracy of the model in botnets is 99.74 percent with an F1-score of 0.98. The CNN-LSTM model exhibits excellent behavior on both the training and test data, as shown in Table 4.

Table 4. Classification score, CNN-LSTM.

	Precision	Recall	F1-Score	Support
Class 0	1.00	0.95	0.98	19,709
Class 1	0.96	1.00	0.98	19,926
Accuracy			0.98	39,635
Macro Avg.	0.98	0.98	0.98	39,635
Weighted Avg.	0.98	0.98	0.98	39,635

4.3. CNN Model Behavior Analysis

The CNN model behavior summary depicted in Figure 12 shows the train–test "loss vs. accuracy" of the model. The test loss decreases steadily at the same interval as the training loss until the sixth epoch. With more iterations, both the training and test loss decreased at a steady rate where the test accuracy exhibits a certain level of steady behavior after the third iteration, contradicting training loss. The CNN model performs steadily during the learning process and behaves as the best model.

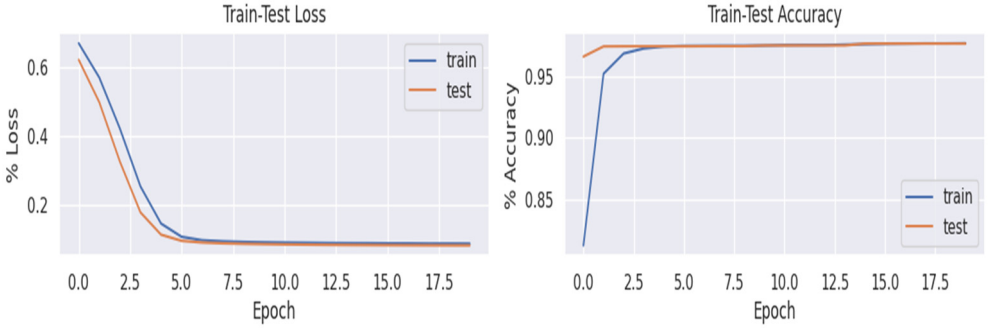

Figure 12. CNN train–test loss vs. accuracy.

Figure 13 describes the CNN model classification score based on "True labels" and "Predicted labels" in confusion matrix findings. The 96.54% accuracy in a normal class is

gained by the model, predicting 19,709 labels of 19,697 true labels. In addition, the model's accuracy forecast for the anomalous class was 99.74%. Based on 19,926 real labels, the model performed with a 98.2% accuracy rate.

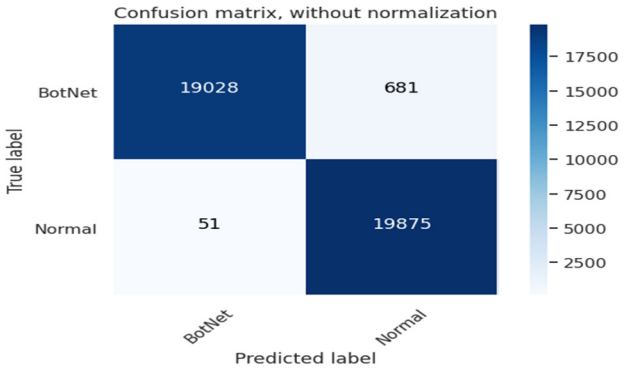

Figure 13. Confusion matrix—CNN.

In both the "Normal" and "Botnet" classes, the CNN model has a recall of 98%. Table 5 shows an overall F1-score of 98%, indicating that CNN performed extremely well during the training and testing phase on the selected dataset.

Table 5. Classification score—CNN.

	Precision	Recall	F1-Score	Support
Class 0	1.00	0.97	0.98	19,709
Class 1	0.97	1.00	0.98	19,926
Accuracy			0.98	39,635
Macro Avg	0.98	0.98	0.98	39,635
Weighted Avg	0.98	0.98	0.98	39,635

5. Discussion

Table 6 shows a "Comparative Analysis" of the models' outcome in terms of the accuracy of suspicious data symptomizing. The data model domain can be described using multiple aspects, that is, the generative models explain how data are placed in the data space and analyzed in an unsupervised way using a probabilistic approach. The discriminative models are swift enough to detect outlier data, predict data, and define boundaries among data classes. The CTS module uses a deterministic approach to solve anomaly detection problems in a supervised way. The better results and behavior predicted by learning algorithms are based on feature engineering and model behavior. From a large number of features, the desirable and relevant attribute selection is able to reduce the data dimension to address the complexity of the problem. The learning of classification models can be improved with dimensional reduction techniques for feature selection [33]. Minimizing the feature size can improve performance, computational efficiency, generalization, and feature interpretability [34]. The redundant and strongly correlated variables cannot assure that the proof of additional information can increase training time [35]. Most of the features are irrelevant (either partially or completely) to the target output, and minimizing irrelevant features can reduce the learning process [36].

In the application context, improvements are made by minimizing the CTS input to 24 Fv for less computational complexity, better response time, and the improved robustness of the model. The unidirectional shallow architecture DBN model [23,29,31], as listed in the following Table, consumes 256 Fv as an input, causing redundant feature extraction with extra training time, the modification of weights, and biases at each learning cycle with more response time and the maximization of the computation. Furthermore, the large size input and the strong correlation of data features can affect output yield.

Table 6. Comparative analysis of past contributions and suggested model performance.

Purposed Model	Architecture	DL Model	Technique	Input FVs Size	Batch Size	Classification	Activation Function	Dataset	Results
Anomaly Detection	Generative	DBN [29]	Sliding Window	256	CPU GPU	Binary	ReLU	CTU-13	99.34%
Anomaly Detection	Generative	DBN [31]	Sliding Window	256	CPU GPU	Binary	ReLU	CTU-13	99.54%
Anomaly Detection	Discriminative	CNN [32]	Image segmentation	3 *	Mini Fixed	Binary	ReLU	CTU-13	72.57%
Anomaly Detection	Discriminative	CNN	Feature-based	24	Fixed	Binary	Leaky ReLU	CTU-13	99.74%

* image input is treated as a single column on each channel (three channels).

The DBN is a greedy model that needs a large number of data features to perform. The ReLU activation function used in the DBN model faces the "dying ReLU" problem, which extends to the training process. To address this limitation, we substituted the ReLU activation with its improved form: the "Leaky ReLU" [37]. The Leaky ReLU can fix the "dying ReLU" concerns with a few slope and leak positive values at "0" to avoid the valuable information vanishing. Its training ability becomes faster when its mean activation reaches close to zero, which is able to increase training efficiency. The DBN follows the backpropagation technique through the modification of weight and fine-tuning in each iteration, as more depth in DBN can cause computational decay. Considering such an issue, for faster convergence, we adopted the Adam (adaptive moments) optimization technique to allow the model operators to learn adaptively with better speed and accuracy. The Adam optimization boosts the model convergence and the learning process, and it also helps to minimize the cost function or loss using operational parameters [38]. Moreover, the categorical cross-entropy loss function is used for encoded input labels (0, 1), ensuring better classification. On the other hand, the CNN [32] model uses the two-dimensional frame input data for image segmentation by transforming time-series data into multiple channels for model learning. The image input utilizes multi-dimensional data as a feature input, which can also decrease response time and increase the computation power of the model in real-time scenarios.

At the next level, the models' interpretation was examined using the "Friedman test" and "post hoc" analysis for significant performances. The non-parametric test was performed to determine the significant difference in botnet classification, and the output metrics of DBN [29,31], CNN [32], and the proposed CNN (CTS) models were analyzed.

The Freidman test value of 9.00 and the related p-value of 0.029 were recorded, revealing different distributions (reject H0). We performed Nemenyi's post hoc test to decide what group of models acted differently. Through the evaluation results, the CNN group of models [32] appear to be statistically significant with different means of 0.022960.

It can be concluded that with fewer feature sizes, the CTS DL module gains relatively better accuracy and performs with higher precision in botnet detection against its counterparts of DBN [29], DBN [31], and CNN [32] without the loss of valuable information carried by same data features. Furthermore, optimization techniques were considered to address the limitations of former models. Improvements have been made with relatively better accuracy and robustness of the model. The proposed CNN (CTS) model performed significantly better than CNN [32] with a high accuracy rate.

6. Conclusions

New tenants are welcomed in 5G networks but this participation can increase the threat level of a 5G core network. Using an adaptive security-based modular approach is beneficial to countering cyber threats in real-time. The KPIs defined in this paper help to deploy the vCTS module on PoPs in the context of 5G network protection. The suggested data filtering and validation approach allows us to undertake the logical separation of a network in the security region. Such security scanners help to share threat information between security NFs, strengthening the defense against expert hackers. Security regions

enable operators to manage better the protection and swift detection of threats by initiating security VNFs on a slice or among slices. DL-based enhanced security NFs and VNFs with multitier approaches provide a comprehensive security view of shared slices and their virtual twins in a shared 5G environment and provide the stress-free integration of updated threat intelligence. Furthermore, comparing and contrasting the DL module to the limitation of past models in 5G network security enables us to provide the best solution in the current context. In the application context, improvements have been made to the input features dimension and computational complexity reduction with better response time and accuracy in outlier detection. The outcomes of all three proposed DL modules were recorded, with the best accuracy rate of 99.74% for botnet classification. For future work, we suggest researching the logical security management of a 5G network in both backhaul and fronthaul communication channels. The shared 5G slices and associated VNFs can be dealt with as "virtual twins" in a shared environment using service-level agreements with 5G tenants.

Author Contributions: Conceptualization, A.M. and A.W.; methodology, A.M., A.W. and S.K.; software, A.M., S.H. and A.K; validation, A.M., S.K. and A.M.A.; formal analysis, A.M., AK., A.M.A., A.N. and S.H.; investigation, A.M.A., S.K., A.K. and A.N.; resources, A.W., A.N., A.M.A. and M.I.; data curation, A.M., A.K. and A.W.; writing—original draft preparation, A.M., A.M.A. and S.K.; writing—review and editing, A.M.A., S.K., A.W., A.K. and S.K.; visualization, A.M., A.W. and A.K.; supervision, A.N. and A.W.; project administration, A.N. and A.W.; funding acquisition, A.M.A. and M.I. All authors have read and agreed to the published version of the manuscript.

Funding: Researchers would like to thank the Deanship of Scientific Research, Qassim University for funding publication of this project.

Data Availability Statement: Our data is embedded in the paper, not separate.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Huawei Technologies, Co. 5G Network Architecture-A High Level View. 2020. Available online: www.huawei.com (accessed on 28 March 2021).
2. Huawei Technologies, Co. 'Huawei: Security Best Practices for 5G. 2021. Available online: www.huawei.com (accessed on 28 March 2021).
3. Suárez, L.; Espes, D.; Le Parc, P.; Cuppens, F.; Bertin, P.; Phan, C.T. Enhancing network slice security via Artificial Intelligence: Challenges and solutions. In Proceedings of the Conférence C&ESAR 2018, Rennes, France, 19–21 November 2018.
4. Bega, D.; Gramaglia, M.; Fiore, M.; Banchs, A.; Costa-Perez, X. DeepCog: Cognitive Network Management in Sliced 5G Networks with Deep Learning. In Proceedings of the IEEE INFOCOM 2019-IEEE Conference on Computer Communications, Paris, France, 29 April–2 May 2019; pp. 280–288. [CrossRef]
5. Palo Alto Networks. Mobile Network Infrastructure Getting Started. 2021. Available online: www.paloaltonetworks.com (accessed on 15 September 2022).
6. Li, X.; Samaka, M.; Chan, H.A.; Bhamare, D.; Gupta, L.; Guo, C.; Jain, R. Network Slicing for 5G: Challenges and Opportunities. *IEEE Internet Comput.* **2018**, *21*, 20–27. [CrossRef]
7. Kotulski, Z.; Nowak, T.; Sepczuk, M.; Tunia, M.; Artych, R.; Bocianiak, K.; Ośko, T.; Wary, J.-P. On end-to-end approach for slice isolation in 5G networks. *Fundam. Chall.* **2017**, *11*, 783–792. [CrossRef]
8. Arfaoui, G.; Vilchez, J.M.S.; Wary, J.-P. Security and Resilience in 5G: Current Challenges and Future Directions. In Proceedings of the 2017 IEEE Trustcom/BigDataSE/ICESS, Sydney, NSW, Australia, 1–4 August 2017; pp. 1010–1015. [CrossRef]
9. Suárez, L.; Espes, D.; Cuppens, F.; Phan, C.-T.; Bertin, P.; Le Parc, P. Managing Secure Inter-slice Communication in 5G Network Slice Chains. In Proceedings of the Data and Applications Security and Privacy XXXIV: 34th Annual IFIP WG 11.3 Conference, DBSec 2020, Regensburg, Germany, 25–26 June 2020; pp. 24–41. [CrossRef]
10. Boutigny, F.; Betgé-Brezetz, S.; Blanc, G.; Lavignotte, A.; Debar, H.; Jmila, H. Solving security constraints for 5G slice embedding: A proof-of-concept. *Comput. Secur.* **2020**, *89*, 101662. [CrossRef]
11. Sattar, D.; Matrawy, A. Towards Secure Slicing: Using Slice Isolation to Mitigate DDoS Attacks on 5G Core Network Slices. In Proceedings of the2019 IEEE Conference on Communications and Network Security (CNS), Washington, DC, USA, 10–12 June 2019; pp. 82–90. [CrossRef]
12. Thantharate, A.; Paropkari, R.; Walunj, V.; Beard, C. DeepSlice: A Deep Learning Approach towards an Efficient and Reliable Network Slicing in 5G Networks. In Proceedings of the 2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), New York, NY, USA, 10–12 October 2019; pp. 0762–0767. [CrossRef]

13. Thantharate, A.; Paropkari, R.; Walunj, V.; Beard, C.; Kankariya, P. Secure5G: A Deep Learning Framework Towards a Secure Network Slicing in 5G and Beyond. In Proceedings of the 2020 10th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 6–8 January 2020; Volume 179, pp. 1–6. [CrossRef]
14. Blanc, G.; Kheir, N.; Ayed, D.; Lefebvre, V.; de Oca, E.M.; Bisson, P. Towards a 5G Security Architecture. In Proceedings of the 13th International Conference on Availability, Reliability and Security, Hamburg, Germany, 27–30 August 2018; p. 47. [CrossRef]
15. Jiang, Z.; Chen, X.; Ma, J.; Zhang, Y.; Gu, J. Traffic Dynamics Evaluation for the Future NFV Deployment. *IEEE Trans. Circuits Syst. II Express Briefs* **2020**, *67*, 2214–2218. [CrossRef]
16. Zhang, C.; Ueng, Y.-L.; Studer, C.; Burg, A. Artificial Intelligence for 5G and Beyond 5G: Implementations, Algorithms, and Optimizations. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **2020**, *10*, 149–163. [CrossRef]
17. Dua, S.; Du, X. *Data Mining and Machine Learning in Cybersecurity*; Taylor & Francis Group: New York, USA, 25 May 2011.
18. Barmpounakis, S.; Magdalinos, P.; Alonistioti, N.; Kaloxylos, A.; Spapis, P.; Zhou, C. Data Analytics for 5G Networks: A Complete Framework for Network Access Selection and Traffic Steering. *Int. J. Adv. Telecommun.* **2018**, *11*. Available online: http://www.iariajournals.org/telecommunications/2018 (accessed on 28 March 2021).
19. Abbas, K.; Khan, T.A.; Afaq, M.; Rivera, J.J.D.; Song, W.-C. Network Data Analytics Function for IBN-based Network Slice Lifecycle Management. In Proceedings of the 2021 22nd Asia-Pacific Network Operations and Management Symposium (APNOMS), Tainan, Taiwan, 8–10 September 2021; pp. 148–153. [CrossRef]
20. Sevgican, S.; Turan, M.; Gokarslan, K.; Yilmaz, H.B.; Tugcu, T. Intelligent network data analytics function in 5G cellular networks using machine learning. *J. Commun. Netw.* **2020**, *22*, 269–280. [CrossRef]
21. Buczak, A.L.; Guven, E. A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 1153–1176. [CrossRef]
22. Vishwakarma, A. Network Traffic Based Botnet Detection Using Machine Learning. Master's Thesis, San Jose State University, Washington, DC, USA, 2020.
23. Hodo, E.; Bellekens, X.; Hamilton, A.; Tachtatzis, C.; Atkinson, R. Shallow and Deep Networks Intrusion Detection System: A Taxonomy and Survey. *arXiv* **2017**, arXiv:1701.02145.
24. Tanhatalab, M.R.; Yousefi, H.; Hosseini, H.M.; Bonab, M.M.; Fakharian, V.; Abarghouei, H. Deep RAN: A Scalable Data-driven platform to Detect Anomalies in Live Cellular Network Using Recurrent Convolutional Neural Network. In Proceedings of the 2020 IEEE 18th World Symposium on Applied Machine Intelligence and Informatics (SAMI), Herlany, Slovakia, 23–25 January 2020; pp. 269–274. [CrossRef]
25. Amanullah, M.A.; Habeeb, R.A.A.; Nasaruddin, F.H.; Gani, A.; Ahmed, E.; Nainar, A.S.M.; Akim, N.M.; Imran, M. Deep learning and big data technologies for IoT security. *Comput. Commun.* **2020**, *151*, 495–517. [CrossRef]
26. Shi, X.; Chen, Z.; Wang, H.; Yeung, D.Y.; Wong, W.K.; Woo, W.C. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 1–9.
27. Chaudhary, H.; Detroja, A.; Prajapati, P.; Shah, P. A review of various challenges in cybersecurity using Artificial Intelligence. In Proceedings of the 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), Thoothukudi, India, 3–5 December 2020; pp. 829–836. [CrossRef]
28. Lee, J.; Kim, J.; Kim, I.; Han, K. Cyber Threat Detection Based on Artificial Neural Networks Using Event Profiles. *IEEE Access* **2019**, *7*, 165607–165626. [CrossRef]
29. Maimo, L.F.; Gomez, A.L.P.; Clemente, F.J.G.; Gil Perez, M.; Perez, G.M. A Self-Adaptive Deep Learning-Based System for Anomaly Detection in 5G Networks. *IEEE Access* **2018**, *6*, 7700–7712. [CrossRef]
30. Maimo, L.F.; Clemente, F.J.G.; Gil Perez, M.; Perez, G.M. On the performance of a deep learning-based anomaly detection system for 5G networks. In Proceedings of the 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), San Francisco, CA, USA, 4–8 August 2017; pp. 1–8. [CrossRef]
31. Maimo, L.F.; Celdrán, A.H.; Clemente, F.J.G. Anomaly Detection on Encrypted and High-Performance Data Networks by Means of Machine Learning Techniques. In *Recent Advances in Security, Privacy, and Trust for Internet of Things (IoT) and Cyber-Physical Systems (CPS)*; Chapman and Hall/CRC: New York, NY, USA, 2020; pp. 167–190. [CrossRef]
32. Doan, M.; Zhang, Z. Deep Learning in 5G Wireless Networks—Anomaly Detections. In Proceedings of the 2020 29th Wireless and Optical Communications Conference (WOCC), Newark, NJ, USA, 1–2 May 2020; IEEE: Piscataway, NJ, USA; pp. 1–6. [CrossRef]
33. Rawat, T.; Khemchandani, V. Feature Engineering (FE) Tools and Techniques for Better Classification Performance. *Int. J. Innov. Eng. Technol.* **2017**, *8*, 169–179. [CrossRef]
34. Srivastava, M.S.; Joshi, M.N.; Gaur, M. A Review Paper on Feature Selection Methodologies and Their Applications. *Int. J. Comput. Sci. Netw. Secur.* **2013**, *7*, 757–761.
35. Iguyon, I.; Elisseeff, A. An introduction to variable and feature selection. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182.
36. Dash, M.; Liu, H. Feature Selection for Classification. *Intell. Data Anal.* **1997**, *1*, 131–156. [CrossRef]

37. Ding, B.; Qian, H.; Zhou, J. Activation functions and their characteristics in deep neural networks. In Proceedings of the 2018 Chinese Control and Decision Conference (CCDC), Shenyang, China, 9–11 June 2018; pp. 1836–1841. [CrossRef]
38. Kingma, D.P.; Ba, J.L. Adam: A Method for Stochastic Optimization; In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015. San Diego, CA, USA, 7–9 May 2015; pp. 1–15.